Biometric Based Cryptographic Key Generation from Faces

B. Chen and V. Chandran
Queensland University of Technology, Brisbane, Qld 4001 AUSTRALIA
brenden_chen@msn.com, v.chandran@qut.edu.au

Abstract

Existing asymmetric encryption algorithms require the storage of the secret private key. Stored keys are often protected by poorly selected user passwords that can either be guessed or obtained through brute force attacks. This is a weak link in the overall encryption system and can potentially compromise the integrity of sensitive data. Combining biometrics with cryptography is seen as a possible solution but any biometric cryptosystem must be able to overcome small variations present between different acquisitions of the same biometric in order to produce consistent keys. This paper discusses a new method which uses an entropy based feature extraction process coupled with Reed-Solomon error correcting codes that can generate deterministic bit-sequences from the output of an iterative one-way transform. The technique is evaluated using 3D face data and is shown to reliably produce keys of suitable length for 128-bit Advanced Encryption Standard (AES).

1. Introduction

Communications advancements in recent decades have led to an increased volume of digital data traveling through publicly shared media. This has led to the rapid development of cryptographic techniques such as AES and public key architectures such as Rivest, Shamir and Adleman (RSA) [1]. Although keys of sufficient length are strong against both brute force and factorization attacks they still suffer from weaknesses due to insecure key protection by user selected passwords. The limitations of passwords are well documented [2, 3]; they are simple and can be easily guessed or obtained using social engineering techniques. They are often written down and stored in an insecure location, can be shared between users, and cannot provide a guarantee of non-repudiation. Furthermore, most people tend to use the same password for a wide range of applications and as a result the compromise of one system leads to the compromise of many others.

In recent years researchers have turned towards merging biometrics with cryptography as a means to improve overall security by eliminating the need for key storage using passwords. During the last decade biometrics has become commonly used for identifying individuals. The success of its application in user authentication has indicated that many advantages could be gained by incorporating biometrics with cryptography. A biometric is an inherent physical or behavioural characteristic of an individual such as their voice, face, fingerprint or keystroke dynamics. Biometrics, in contrast to passwords, cannot be forgotten, are difficult to copy or forge, impossible to share and offer more security then a common eight character password.

The principal drawback of a biometric is that it is not precise like a password or cryptographic key. Limitations of acquisition technology and the inherent changes in the biometric (such as pose and expression for faces) and environmental conditions (such as lighting) lead to variations in each sample of the same biometric. For example, although an iris is considered to be the most accurate of biometrics, there can be up to 30% variation between two different images of the same iris [4]. It is the primary challenge of all biometric cryptosystems to overcome this variation whilst harnessing the advantages of biometrics in order to improve the security of encryption keys.

Another challenge stems from the permanence of a biometric. Apart from physical damage, fingerprints or iris remain largely unchanged throughout a person's life. This is a desired property in most applications of biometrics but in cryptography this is a weakness. Cryptographic keys need to be (and they often are) revoked or changed both proactively as a measure to increase security and reactively as response to key compromise. Most proposed schemes ultimately come down to the protection of an existing cryptographic key with biometric information. While the existing key can

be changed the biometric used to secure it cannot and this shortcoming is often neglected.

This paper discusses how keys can be generated using a biometric and demonstrates the technique using 3D face images. It shows how a biometric based binary sequence (bio-key) can be generated by selecting bits from the binary representation of the output of a chaotic bispectral transform applied to the user biometric. The transform is designed to eliminate bit errors arising from similarity transformations of the biometric and yet be sensitive to small changes. The length of these bio-keys can be increased as desired and they can be changed (i.e. revoked) by altering the procedure. Due to the fuzzy nature of biometrics these bio-keys contain some bit errors and as such cannot be directly used as cryptographic keys. But an error correction method similar to those used by Monrose [5, 6], Juels [7] and Hao [8] allows these biometric based sequences to be used in regenerating an existing cryptographic key exactly. Performance of the system is analyzed using false accept and false reject statistics and bit error distributions.

The underlying bispectral transform, binary conversion and bit selection criteria have been previously discussed in detail in [9] where the transform was used to generate random numbers from 3D face images but will be briefly described again for clarity in section 3.

2. Background

Most recent biometric cryptosystems can be grouped into two categories. The first category relies on Shamir's secret share scheme [10]. This method is based on the use of polynomial interpolation to completely reproduce a polynomial given a small set of coordinate points (also known as shares) that lie on the polynomial. The cryptographic key can be hidden in this polynomial, either split up into the coefficients or as the constant term. The user's biometric data is then used to determine which shares are used to reproduce the polynomial and recover the key.

The first use of this technique was by Monrose et al. [5] and presented as a technique of password hardening based on keystroke dynamics. This was expanded upon later by the same investigators using voice features [6]. The method protects a key with a number of arbitrarily created shares, one share for each feature extracted from the user's voice. These shares are placed in a two column lookup table. A threshold is applied to each of the user's voice features and depending on the decision a share is selected from either the left or the right column of the lookup table. This method provided 60 bits of security. Other research groups have since applied this method to other biometrics such as fingerprint [11, 12], face [13] and iris [8]. However, the majority of these methods, with the exception of Hao et al.[8] who can produce 140 bits, have at best only 88 bits of entropy. This is only marginally more than *half* of the entropy in a modern day 128-bit AES key that the schemes are intended to protect.

Juels and Sudan [14] proposed their Fuzzy Vault scheme which also uses polynomial interpolation similar to Shamir's secret share. But unlike the method proposed by Monrose, where biometric information is used to search the look-up table, this method uses biometric data to create the shares (coordinate points) themselves. Again, only a subset of the coordinate pairs are required to recreate the polynomial in order to obtain the key. A large number of chaff points, which are randomly generated points that do not lie on the polynomial, are also inserted into the stored template so an attacker without the biometric cannot generate the key. The use of polynomial coordinate points makes this method more suited to biometrics that suffer from incorrect ordering and erasures between acquisitions.

Fuzzy Vaults have been used by a number of researchers in their implementations of biometric cryptography systems. Clancy et al. [15] used fingerprint data which demonstrated the feature order invariance of the Fuzzy Vault scheme. Reed-Solomon codes are used to aid in polynomial interpolation to recover the key. The system was shown to produce 69 bits of complexity and simulated performance indicated a false reject rate (FRR) of 20-30% at a theoretical false accept rate (FAR) of 2^{-69}. Uludag et al. [16] proposed a very similar method also using fingerprints which produced roughly 35 bits of security and empirical testing showed a FRR of 21% at 0% FAR. An identical fuzzy vault technique applied to handwritten signatures by Kholmatov and Yanikoglu [17] obtained FRR of 8.3% with FAR of 2.5%.

In summary, most techniques can only generate short keys that are of insufficient length for modern cryptography protocols. Secondly, there is little or no discussion on how to handle the event when a biometric is compromised and can longer be used to protect keys. High FRR also limits the use of these techniques. For certain applications and biometric modalities high FRR may, however, not be a serious problem. An example is faces from a video stream.. A false rejection rate of 95% will result in the need for 20 frames on average before a frame is accepted and this is still a delay of less than a second at 25 frames per second.

3. Proposed method

The transform employed by the system is an iterative, chaotic, bispectral one-way transform [18] that accepts a one-dimensional vector input and is used to produce a magnitude and angle pair per iteration. The transform incorporates similarity transformation invariance and shape sensitivity by design. This output can be converted to binary to form a very large bit matrix. These matrices are analyzed to locate feature bits suitable to be used as part of the bio-key using an entropy based criteria described in section 3.2.

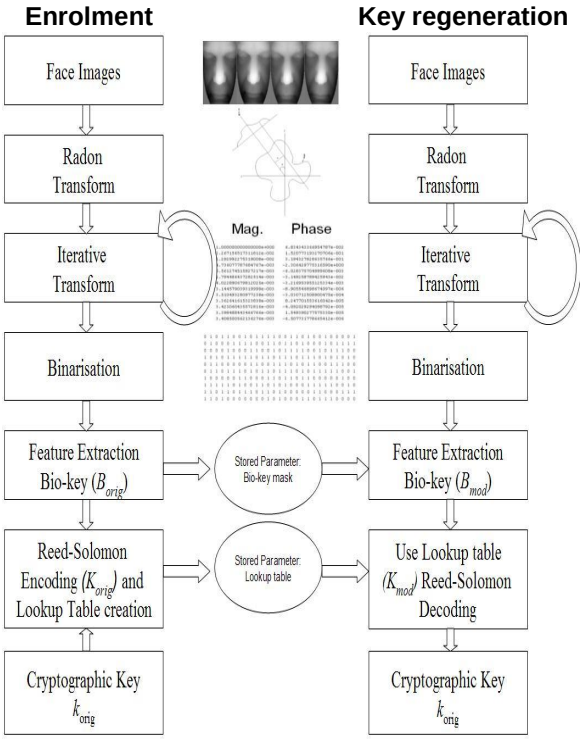

Figure 1. Biometric cryptosystem flowchart

3.1 The transform

The transform requires a 1D input vectors and the Radon transform is used to convert a 2D image into a set of 1D projections. The resulting 1D vector produced at each rotation angle is fed into the bispectral transform. This vector is first normalized by the magnitude of the largest vector element; the mean is also removed. This normalised vector ($x_i(n)$), where i denotes the i-th iteration) is then N-point Discrete Fourier Transformed to obtain $X_i(k)$ from which the magnitude spectrum ($|X_i(k)|$) is computed and the negative half of the frequency spectrum is discarded.

To make the transform a one-way irreversible process the Fourier phase information is discarded. The sequence is then zero-padded to length N to produce a real valued sequence:

$$y_i(n) = \begin{cases} |X_i(n)|, n = 1,2,..., N/2-1 \\ 0, \quad n = N/2,...N-1 \end{cases}$$

The imaginary part is set to zero. The Fourier Transform is applied to this sequence to produce the bispectrum: $B_i(k_1,k_2) = Y_i(k_1)Y_i(k_2)Y_i^*(k_1,k_2)$ where * represents complex conjugation. Unlike $|X_i(k)|$ the phase information is not discarded from the bispectrum. Therefore the bispectrum is complex valued with non-zero imaginary components and is sensitive to asymmetry.

The bispectrum is then integrated along radial slices in the bifrequency plane to obtain: $V_i(a) = \int_{k_1=0+}^{1/(1+a)} B_i(k_1, ak_1)dk_1$ where $a = \frac{1}{N}, \frac{2}{N}, ..., 1$.

Frequencies are normalized by one half of the sampling frequency (Nyquist frequency). The zero frequency component (or average signal) is eliminated from the above computation and a is the slope of the line in bifrequency (k_1, k_2) space along which the integral is computed. The bispectrum is bilinearly interpolated so the integration can be computed using summation. The application and properties of this procedure for feature extraction have been described previously by Chandran and Elgar [18].

Iterating the procedure is necessary to produce bit-sequences since more iterations increases the size of the output and the therefore enlarges the potential pool of bits [9]. The transform can be easily modified to become iterative by feeding back the integrated bispectrum as a complex valued input vector of length N for the next iteration. The normalisation step applied guarantees the system will be BIBO stable regardless of the number of iterations taken.

After each iteration of the procedure, a measure of change is extracted by computing the complex valued inner product of the difference between the previous and present outputs with the previous output to obtain:

$$D_i(n) = \sum_{n=0}^{N-1} [x_{i-1}(n) - x_i(n)] x_{i-1}(n) = M_i \exp(j\phi_i).$$

Where D represents the difference and can be represented as a Magnitude (M) and Phase (Φ) pair (one pair per iteration) which can be used to form a Magnitude/Phase matrix.

3.2 Binary feature extraction

These magnitude and angle matrices must then be binarised to allow for entropy calculation in order to determine desirability of the bits. The base 2 logarithm of the magnitude and angle values are stored as 64 bit floating point representation in the standard IEEE format [19]. These binary sequences are then stored in two matrices, one for the angle and the other for magnitude.

Once the matrices are converted into binary form it is important that a definite quantitative process is developed to identify the suitable feature bits. To do this we must first determine what exactly a "desirable bit" is, then identify a statistical property that can be used to quantify this desirability. A method must also be developed that allows the bits to be ranked from the most desirable to the least. The use of statistical properties requires a training set data. For each user the training set can be logically split into two parts, an intra-class set (matrices generated from images of this user) and an extra-class set (matrices generated using images of all other users).

Bit probability seems like a natural statistical property that can be used to describe a bit's desirability. However bit probability is is dual valued in this context; a bit probability of 0 or 1 represents the same level of constancy, likewise bit probabilities of 0.49 or 0.51 represent the same level of randomness. Hence binary entropy is preferred. Entropy measures the amount of information contained in a bit on a scale from zero to one. The two extreme values of 0 and 1 correspond to a constant bit (no information) and a completely random bit (one bit information), respectively. The desirability of a bit as a feature and as part of the bio-key can be represented using its intra and extra class entropies. This criterion is represented in Table 1.

Rarely will a bit fit perfectly into any of the above categories. Instead each bit can then be given a weight value between 0 and 1 depending on its usefulness allowing for the quantitative ranking of bits in order of their desirability as a feature bit. The weight (w) is calculated using a function based on both the intra and extra class entropies (η_{intra} and η_{extra}) and can be broken up into two parts:

1. Intra-class weight: from Table 1 it can be seen that ideal intra class entropy should be low and these bits should have a high weighting.

$$w_1 = 1 - \eta_{\text{intra}}$$

2. Extra-class weight: should be high when extra class entropy is high.

$$w_2 = \eta_{\text{extra}}$$

The overall weight of a bit is simply the product of the intra-class weight and the extra-class weight.

The N highest weighted bits can then be used to form an N-bit bio-key, the locations of the N highest weighted bits are stored and used as a mask that can be applied to the bit matrix derived from future presentations of the biometric in order to extract the same N-bit bio-key.

Table 1. Bit properties and the corresponding value for key generation

	Low Intra Class Entropy	**High Intra Class Entropy**
High Extra Class Entropy	Desirable	Undesirable
Low Extra Class Entropy	Undesirable	Undesirable

The use of the Radon transform coupled with the iterative nature of the bispectral transform and the systematic process of bit extraction means that the same biometric can produce many different N-bit bio-keys. This can be done by altering the number of rotations or iterations taken by the transform or by limiting where the bits that form the bio-key can come from (i.e. use only bits from even rotations, or odd iterations, or only from magnitude or any combination of). These parameters can be stored on a smartcard and even if the biometric is compromised the bio-key cannot be reproduced without these parameters. Even if both were to be lost a new bio-key can be issued using new parameters. The parameters act like a user chosen 'password' that is written into the smart card but cannot be directly used by an attacker without possession of the card.

3.3 Error correction

Error correction is performed in a manner similar to Monrose [5] in that a two column look up table is used. But instead of Shamir's secret share, Reed-Solomon (RS) codes are used to correct errors similar to Juels [7] and Hao [8]. The lookup table is created at enrolment by first taking the original k-bit

cryptographic key (k_{orig}) that needs to be secured. This is encoded with a RS coding scheme to create an N-bit sequence (K_{orig}). The N-bit bio-key (B_{orig}) is generated using the training images of the user. These two N-bit sequences (K_{orig} and B_{orig}) are combined to form a two column lookup table of N rows. These three parameters (k_{orig}, K_{orig} and B_{orig}) can then be discarded and only the lookup table is retained. Future acquisitions of the same biometric will produce a slightly modified bio-key B_{mod} when applied to the lookup table will extract K_{mod}. The number of errors between B_{mod} and B_{orig} will be same as K_{mod} and K_{orig} and if this is within the decoding ability of the RS code used then K_{mod} can be successfully decoded to produce the original cryptographic key k_{orig} without error.

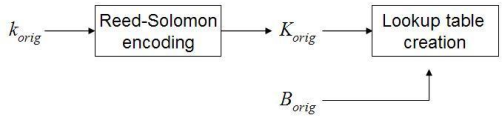

Figure 2. Creation of lookup table during enrolment

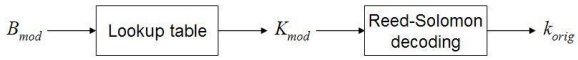

Figure 3. Regeneration of k_{orig} from using B_{mod} and lookup table

This concept is illustrated in the example below assuming for simplicity that k_{orig} = 11, once encoded becomes K_{orig} = 1101 and the user's bio-key is B_{orig} = 1010 the resulting lookup table would be (where B_n is the n-th bit in B_{orig}):

	If B_n is 0	If B_n is 1
n=1	0	**1**
n=2	**1**	0
n=3	1	**0**
n=4	**1**	0

As can be seen the bits in K_{orig} (bolded) have been placed into the table depending on the corresponding bit in B_{orig}. On a different presentation of the biometric by the same user a slightly altered bio-key B_{mod} = 1011 is produced and when applied to lookup table it will generate K_{mod} = 1100. The one bit error in B_{mod} translates to a one bit in error in K_{mod} relative to B_{orig} and K_{orig}. Decoding of K_{mod} will correct this one bit error returning the original k_{orig} = 11.

The use of the intermediate variable K_{mod} and the lookup table may seem like an unnecessary complexity when it seems possible to apply Reed-Solomon decoding directly on the bio-key (B_{orig}) in order to produce an error free binary sequence suitable for use as a cryptographic key. But there are two main reasons for not using this simpler approach:

1. B_{orig} is a random arbitrary sequence and may be unsuitable for Reed-Solomon decoding. The use of the lookup table can be seen as mapping a possibly undecodable sequence B_{orig} to a decodable sequence K_{orig} (K_{orig} is always decodable since it is the output of a Reed-Solomon encoder).

2. Directly generating a cryptographic key from the bio-key would mean the method is unable to protect already existing keys. The random arbitrary nature of B_{orig} also means that a derived key may lack performance or security properties that an existing key would have.

3.4 Security analysis

To successfully unlock the cryptographic key (k_{orig}) an attacker would require:

- The user biometric.
- Lookup table
- Transform parameters for bio-key

The compromise of any one of these factors only is not enough to aid an attacker in the regeneration of k_{orig}. The user biometric cannot be used to generate the bio-key without the stored parameters (contained either in a smartcard or central database store); alternatively the theft of the stored parameters alone is also of no help. Even if stored parameters are known to the attacker no information about k_{orig} can be extracted. The lookup table used to produce K_{mod} is derived through the combination of k_{orig} and B_{orig} which are generated independently from two completely separate random processes, k_{orig} produced from any modern cryptographically secure random number generator and B_{orig} produced from the bispectral transform whose outputs have been shown to pass standard National Institute of Standards and Technology (NIST) tests for statistical randomness [9].

With the knowledge of the lookup table the attacker could theoretically attempt to brute force B_{orig} and since the error correction method used tolerates errors in B_{orig} an attacker could successfully reproduce k_{orig} with any value B_{mod} within a certain Hamming distance of B_{orig} , this exact distance is dependent on the RS coding scheme used. However B_{orig} is a longer binary

sequence than k_{orig} itself and Hao [8] has shown that brute forcing B_{orig} is no simpler than brute forcing k_{orig} directly.

Brute forcing B_{orig} is similar to exhaustively trying every possible combination of the lookup table and this is one of the advantages that using binary features affords compared to the original method by Monrose [5]. The original methods by Monrose used keystroke dynamics and voice and were severely limited by the amount of feature information contained in these biometric modalities. This lack of useable features restricts the size of lookup tables and this is the reason why earlier works by Monrose could only produce 16 [5] and 60 bits [6] of security. The method described in this paper overcomes this by mapping the input biometric into a very large matrix of binary bits. However, correlations and dependencies must exist between these bits and not all carry useful feature information but the use of entropy based analysis is a systematic and quantitative method of identifying and extracting the ones that do.

4. Implementation and results

To test the method 3D images from the Face Recognition Grand Challenge (FRGC) Database [20] were used. The images from this database contain many users under various expressions as shown in figure 2. A normalisation procedure is applied to each image by firstly rotating the image to align the eyes followed by image scaling to create uniform eye to eye distance of 70 pixels for all images. A circular mask is then applied to zero out pixels that do not lie in the face, this helps focus the transform on actual features rather than elements of the background. The image is smoothed using histogram equalization prior to being normalized so the non-masked pixels have mean zero and standard deviation of one. Inconsistent data points in the form of spikes and holes created due to sensor noise in acquisition equipment are eliminated using median filtering and linear interpolation, resulting in a final output image of size 150x130 pixels. A subset of this database containing 61 users with a total number of 1417 images was used to train and test the system. The use of entropy as a means of identifying features requires training over a large number of images in order to get reliable estimates of entropy. So 19 images from each user were used to train with and the remainder used in testing, on average each user had 4 testing images.

The Radon transform is used to reduce each 2D image into 90 separate one-dimensional vectors obtained by taking 90 rotations of the image equally spaced between angles from 0 to 180. Each of these 1D vectors is passed through twenty five iterations of the bispectral transform producing 90 angle/magnitude matrices. These are then binarised to form an overall matrix containing 288,000 bits from which good feature bits can be identified and used to generate bio-keys of various lengths.

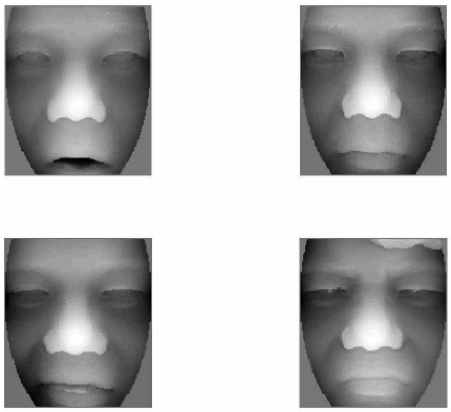

Figure 4. Expression changes present in 3D FRGC face images

The table below shows the average variations contained in bio-keys of varying lengths, it also presents the false accept (FA) and false reject (FR) performance of a biometric cryptographic scheme that uses these bio-keys directly without applying error correction. As can be seen the false reject rates are quite high and are impractical for any application. However the average variation present in the bio-keys is about 6% and the vast majority of true user cases produce fewer than the average number of errors where as an impostor on average will produce a bio-key where half the number of bits are incorrect. This is the result of the entropic feature extraction criteria that seeks to minimize intra class variation while promoting random extra class behavior. These error patterns suggest that the use of error correction can significantly improve the FR performance.

Table 2. Performance of bio-keys (B) of varying lengths

N	False Accept	False Reject	Mean bit error	Impostor Mean bit error
64	0.26%	66%	4.6	31.6
128	0.08%	76%	8.2	62
256	0.01%	88%	15.5	118
512	0.0006%	95%	30	219

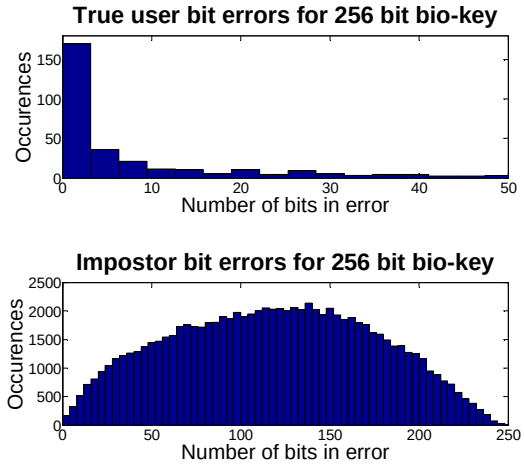

Figure 5. Bit error histograms for true users and impostors

Bio-keys of length 240 bits were chosen and RS encoding scheme using a 4 bit codeword and 15 codeword blocks were selected. To allow for varying length keys (k_{orig}) the RS encoding scheme is altered to correct a different number of blocks (T).

Table 3. Performance of system for varying crypto-key (k_{orig}) lengths using 240 bit bio-key (B_{orig})

T	*k*	False Accept	False Reject	Mean bit error	Impostor Mean bit error
4	112	1.22%	28%	6	63
3	144	0.75%	37%	7.3	68
2	176	0.34%	47%	10	83
1	208	0.09%	63%	13.8	98

The use of error correction provides a tradeoff between false accept and false reject. Although the number of false rejects have been lowered (at the expense of false accepts) they still remain quite high. This is a limiting factor for most biometrics except for 2D face video where multiple frames can be captured each second. False accepts are also relatively high but it is important to keep in mind that a false accept can only occur when the impostor has access to the genuine user's stored parameters which can be kept on tamper proof smartcards. Performance is also expected to improve with the implementation of more rigorous error correction such as majority coding, which can be applied directly to the bio-keys prior to using the lookup table. Preliminary tests have shown this to lower both the false accept and false reject performance of the bio-key. With the false reject performance improving by as much as 14% and the mean bit error falling by as much as 50%. The incorporation of Hadamard coding, which is better suited to correcting single bit errors, prior to Reed-Solomon coding is also expected to improve overall FR performance.

5. Conclusion

This paper illustrates a method of securing a cryptographic key of arbitrary length using a given biometric. Although other biometric based methods have been proposed that have superior FRR/FAR performance few can produce keys of this length. The method is also flexible, the bio-keys used to protect the cryptographic key can be changed and revoked and is a significant feature not possessed by other methods. The method can be modified to protect keys of increasing length by either increasing the size of bio-keys through performing more rotations/iterations of the bispectral transform or by changing the RS encoding scheme used. Lastly, the method can theoretically be applied to any biometric as multidimensional biometrics can be reduced to one dimensional projections or feature vectors.

6. References

[1] A. Menezes, P. van Oorschot and S. Vanstone, Handbook of Applied Cryptography, USA: CRC Press, pp 180, 1997.

[2] D. V. Klein, "Foiling the Cracker: A Survey of, and Improvements to, Password Security," in Proceedings of the 2nd USENIX UNIX Security Workshop, pp. 5-14, 1990.

[3] K.-P. L. Vu, R. W. Proctor, A. Bhargav-Spantzel, B.-L. Tai, J. Cook, and E. Eugene Schultz, "Improving password security and memorability to protect personal and organizational information," International Journal of Human-Computer Studies, vol. 65, pp. 744-757, 2007.

[4] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain, "Biometric cryptosystems: issues and challenges," Proceedings of the IEEE, vol. 92, pp. 948-960, 2004.

[5] F. Monrose, M. K. Reiter, and S. Wetzel, "Password hardening based on keystroke dynamics," Conference on Computer and Communications Security, 1999.

[6] F. Monrose, M. K. Reiter, L. Qi, and S. Wetzel, "Cryptographic key generation from voice," in Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on, pp. 202-213, 2001.

[7] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in Proceedings of the 6th ACM conference on

Computer and communications security, Kent Ridge Digital Labs, Singapore, pp. 28-36, 1999.

[8] F. Hao, R. Anderson, and J. Daugman, "Combining Crypto with Biometrics Effectively," IEEE Transactions on Computers, vol. 55, pp. 1081-1088, 2006.

[9] V. Chandran and B. Chen, "Simultaneous biometric verification and random number generation," in 5th Workshop on the Internet, Telecommunications and Signal Processing WITSP'06, 2006.

[10] A. Shamir, "How to share a secret," Commun. ACM, vol. 22, pp. 612-613, 1979.

[11] M. Y. Siyal and F. Ahmed, "A biometric-based scheme for enhancing security of cryptographic keys," in 2004 IEEE Region 10 Conference TENCON 2004., Vol. 2, pp. 407-410, 2004.

[12] F. Ahmed and M. Y. Siyal, "A novel approach for regenerating a private key using password, fingerprint and smart card," Information Management & Computer Security, vol. 13, p. 39, 2005.

[13] D. C. L. Ngo, A. B. J. Teoh, and A. Goh, "Biometric hash: high-confidence face recognition," Circuits and Systems for Video Technology, IEEE Transactions on, vol. 16, pp. 771-775, 2006.

[14] A. Juels and M. Sudan, "A fuzzy vault scheme," in Proceedings of IEEE International Symposium on Information Theory, p. 408, 2002.

[15] T. C. Clancy, K. Negar, and J. L. Dennis, "Secure smartcard based fingerprint authentication," in Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications, Berkley, California, pp. 45-52, 2003.

[16] U. Uludag, S. Pankanti, and A. K. Jain, "Fuzzy Vault for Fingerprints," in Audio- and Video-Based Biometric Person Authentication, pp. 310-319, 2005.

[17] A. Kholmatov and B. Yanikoglu, "Biometric Cryptosystem Using Online Signatures," in Computer and Information Sciences – ISCIS 2006, pp. 981-990, 2006.

[18] V. Chandran and S. L. Elgar, "Pattern Recognition Using Invariants Defined From Higher Order Spectra- One Dimensional Inputs," IEEE Transactions on Signal Processing, vol. 41, p. 205, 1993.

[19] IEEE Std 754-1985, Standard for Binary Floating Point Arithmetic, Section 3 – Formats.

[20] P. J. Phillips, P. J. Flynn, T. Scruggs, Kevin W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the face recognition grand challenge,” in Proc. of IEEE Comp. Society Conf. on Computer Vision and Pattern Recognition (CVPR’05)-Vol. 1,Washington, DC, USA, pp. 947–954, 2005.